

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE  
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

In re Application of:	§	Filed: March 18, 2004
Dettinger et al.	§	
	§	Group Art Unit: 2162
Serial No.: 10/803,603	§	
	§	Examiner: Giovanna B. Colan
Confirmation No.: 5874	§	

For: A METHOD FOR PROVIDING WORKFLOW FUNCTIONALITY AND  
TRACKING IN AN ANNOTATION SUBSYSTEM

MAIL STOP APPEAL BRIEF - PATENTS  
Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

**CERTIFICATE OF MAILING OR TRANSMISSION**

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Mail Stop Appeal Brief - Patents, Commissioner for Patents, P. O. Box 1450, Alexandria, VA 22313-1450, or facsimile transmitted to the U.S. Patent and Trademark Office to fax number 571-273-8300 to the attention of Examiner Giovanna B. Colan, or electronically transmitted via EFS-Web, on the date shown below:

March 17, 2008	/Karla Marquez/
Date	Karla Marquez

Dear Sir:

**APPEAL BRIEF**

Applicants submit this Appeal Brief to the Board of Patent Appeals and Interferences on appeal from the decision of the Examiner of Group Art Unit 2162 dated September 21, 2007, finally rejecting claims 1, 4-21 and 24-31. The final rejection of claims 1, 4-21 and 24-31 is appealed. This Appeal Brief is believed to be timely since it is transmitted by the due date of March 17, 2008, as set by the filing of a Notice of Appeal on January 16, 2008.

Please charge the fee of \$510.00 for filing this brief to:

Deposit Account No. 09-0465 / ROC920030329US1.

**TABLE OF CONTENTS**

1.	Identification Page.....	1
2.	Table of Contents .....	2
3.	Real Party in Interest .....	3
4.	Related Appeals and Interferences .....	4
5.	Status of Claims .....	5
6.	Status of Amendments .....	6
7.	Summary of Claimed Subject Matter .....	7
8.	Grounds of Rejection to be Reviewed on Appeal .....	11
9.	Arguments .....	12
10.	Conclusion .....	15
11.	Claims Appendix .....	16
12.	Evidence Appendix .....	21
13.	Related Proceedings Appendix .....	22

### **Real Party in Interest**

The present application has been assigned to International Business Machines Corporation, Armonk, New York.

### **Related Appeals and Interferences**

Applicant asserts that no other appeals or interferences are known to the Applicant, the Applicant's legal representative, or assignee which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

### **Status of Claims**

Claims 1, 4-21 and 24-31 are pending in the application. Claims 1-31 were originally presented in the application. Claims 2-3 and 22-23 have been canceled without prejudice. Claims 1, 4-21 and 24-31 stand finally rejected as discussed below. The final rejections of claims 1, 4-21 and 24-31 are appealed. The pending claims are shown in the attached Claims Appendix.

### **Status of Amendments**

All claim amendments prior to the Final office Action have been entered by the Examiner. Proposed amendments to the claims after the final rejection were not entered.

## Summary of Claimed Subject Matter

### A. CLAIM 1 – INDEPENDENT

In the embodiments of independent claim 1, a method for tracking the status of an annotation is provided (See for example, *Application*, FIGs. 4, 6). The method comprises creating an annotation record 150 comprising one or more fields 153 for storing annotation data comprising the annotation (See for example, *Application*, para. [0037-0038], FIGs. 1, 2, 4), retrieving annotation data stored in the annotation record 150 (See for example, *Application*, para. [0046]:2-4, [0047]:5-8, [0059]:7-9, FIGs. 2, 6 (step 604)), applying a set of state rules 170 to determine a first state 400<sub>1-6</sub> of the annotation based on the annotation data (See for example, *Application*, para. [0040]:5-7, [0046], [0047]:5-8, [0059]:8-9, FIGs. 2, 4, 6 (step 606)), receiving additional annotation data (See for example, *Application*, para. [0052]:3-5, [0053], FIG. 5A-5D), updating the annotation record 150 with the additional annotation data (See for example, *Application*, para. [0052], [0053], FIG. 5A-5D), applying the set of state rules 170 to determine a second state 400<sub>1-6</sub> of the annotation based on the annotation data in the updated annotation record 150 (See for example, *Application*, para. [0040]:5-7, [0046], [0047]:5-8, [0059]:8-9, FIGs. 2, 4, 6 (step 606)), and providing an indication that the state of the annotation has changed from the first state 400<sub>1-6</sub> to the second state 400<sub>1-6</sub> (See for example, *Application*, para. [0048], FIG. 4).

### B. CLAIM 13 - INDEPENDENT

In the embodiments of independent claim 13, a method for managing annotations having multiple states 400<sub>1-6</sub>, is provided (See for example, *Application*, FIG. 4). The method comprises defining a plurality of annotation types (See for example, *Application*, para. [0038]:1-3, FIG. 1, FIGs. 4, 5A-5D (illustrating a “problem tracking” annotation type)), each annotation type having one or more associated fields 153 (See for example, *Application*, para. [0038]:3-7, FIGs. 1, 2, 4, 5A-5D (illustrating a “problem tracking” annotation type)), defining a set of state rules 170 for each annotation type (See for example, *Application*, para. [0040], FIGs. 1, 2), wherein each state rule 170 identifies an annotation state based on annotation data in the one or more fields 153

associated with its corresponding annotation type (See for example, *Application*, para. [0046], FIGs. 2, 3 (step 306)), and providing a state machine 172 capable of retrieving annotation data for an annotation of one of the defined annotation types (See for example, *Application*, para. [0046], FIGs. 2, 3 (step 306)), applying the state rules 170 for that type to the annotation data to determine the state 154 of the annotation (See for example, *Application*, para. [0046], FIGs. 2, 3 (step 306)), and providing an indication of the annotation state 154 (See for example, *Application*, para. [0048], FIGs. 2, 4).

#### C. CLAIM 18 - INDEPENDENT

In the embodiments of independent claim 18, a method for gathering information about a plurality of processes of a similar process type is provided (See for example, *Application*, para. [0050], FIGs. 4 (illustrating a “problem tracking” process), 6). The method comprises providing an annotation form 500 for receiving annotation data in a plurality of fields 502, 504, 505 related to the processes (See for example, *Application*, para. [0050], [0051], FIG. 5), storing annotation data received via the annotation form 500 in a plurality of annotation records 150 (See for example, *Application*, para. [0047], [0059], FIGs. 2, 5, 6 (step 604)), wherein each annotation record 150 relates to one of the similar type processes (See for example, *Application*, para. [0047], [0059]:6-7, FIG. 2), providing a set of state rules 170 defining a plurality of states 154 for the annotation based on the annotation data in each record 150 (See for example, *Application*, para. [0040]:3-5, [0059], FIG. 2), applying the state rules 170 to the annotation data in each record 150 to determine the state 154 of each annotation (See for example, *Application*, para. [0059]:8-9, FIGs. 2, 6 (step 606)), and generating a report 700 indicating the state 154 of each annotation (See for example, *Application*, para. [0059]:10, FIGs. 2, 6 (step 608), 7).

#### D. CLAIM 21 - INDEPENDENT

In the embodiments of independent claim 21, a computer-readable storage medium containing a program for managing annotations having multiple states 400<sub>1-6</sub> which, when executed by a processor 130, performs operations is provided (See for example, *Application*, FIGs. 1, 2). The operations comprise creating an annotation



record 150 comprising one or more fields 153 for storing annotation data (See for example, *Application*, para. [0037-0038], FIGs. 1, 2, 4), retrieving annotation data stored in the annotation record 150 (See for example, *Application*, para. [0046]:2-4, [0047]:5-8, [0059]:7-9, FIGs. 2, 6 (step 604)), applying a set of state rules 170 to determine a first state 400<sub>1-6</sub> of the annotation based on the annotation data (See for example, *Application*, para. [0040]:5-7, [0046], [0047]:5-8, [0059]:8-9, FIGs. 2, 4, 6 (step 606)), providing an indication of the first state 400<sub>1-6</sub> of the annotation in an interface screen 500 displaying the annotation data (See for example, *Application*, para. [0051]:6-7, FIG. 5A), receiving additional annotation data (See for example, *Application*, para. [0052]:3-5, [0053], FIG. 5A-5D), updating the annotation record 150 with the additional annotation data (See for example, *Application*, para. [0052], [0053], FIG. 5A-5D), applying the set of state rules 170 to determine a second state 400<sub>1-6</sub> of the annotation based on the annotation data in the updated annotation record 150 (See for example, *Application*, para. [0040]:5-7, [0046], [0047]:5-8, [0059]:8-9, FIGs. 2, 4, 6 (step 606)), and providing an indication of the second state 400<sub>1-6</sub> of the annotation in an interface screen 500 displaying the annotation data (See for example, *Application*, para. [0054]:4-5, [0055]:1-3, FIGs. 5B-5D).

#### E. CLAIM 26 - INDEPENDENT

In the embodiments of independent claim 26, an annotation system 100 is provided (See for example, *Application*, FIG. 1). The system 100 comprises a processor 130, one or more annotation structures 149, each identifying one or more annotation fields associated with an annotation type (See for example, *Application*, para. [0038]:1-3, FIG. 2), an annotation store 139 for storing annotation records 150 (See for example, *Application*, para. [0037]:7-8, FIG. 2), each having fields 153 associated with one of the annotation types (See for example, *Application*, para. [0037-0038], FIGs. 1, 2, 4), a set of state rules 170 for each annotation type, wherein each set of state rules 170 defines a plurality of states 154 for each associated annotation type based on the annotation data in the one or more associated fields 153 (See for example, *Application*, para. [0045], FIGs. 1, 2), and a state machine 172 which, when executed by the process, is configured to access an annotation record 150 and apply the set of state rules 170 for

the corresponding annotation type to determine an annotation state 154 based on the date stored therein (See for example, *Application*, para. [0040]:5-7, [0046], [0047]:5-8, [0059]:8-9, para.[0062], FIGs. 2, 4, 6 (step 606)).

### **Grounds of Rejection to be Reviewed on Appeal**

1. Rejection of claims 1, 4-21 and 24-31 under 35 U.S. 103(a) as being unpatentable over *Bays et al.* (U.S. Patent No. 6,519,603, hereinafter “*Bays*”) in view of *Setya* (U.S. Patent Pub. No. 2006/0111953).

## **ARGUMENTS**

### **1. The combination of *Bays* in view of *Setya* Does Not Render Claims 1, 4-21 and 24-31 Obvious.**

Claims 1, 4-21 and 24-31 stand rejected under 35 U.S.C. 103(a) as being unpatentable over *Bays* in view of *Setya*.

#### *The Applicable Law*

Applicants respectfully traverse this rejection based on the grounds that the Examiner has not established a case of *prima facie* obviousness. The Examiner bears the initial burden of establishing a *prima facie* case of obviousness. See MPEP § 2142. The Federal Circuit has recently held that if all of the elements of a claimed invention are found in a combination of prior art references, analysis requires “consideration of two factors:

(1) whether the prior art would have suggested to those of ordinary skill in the art that they should make the claimed composition or device, or carry out the claimed process; and

(2) whether the prior art would also have revealed that in so making or carrying out, those of ordinary skill would have a reasonable expectation of success.” *PharmaStem Therapeutics, Inc. v. ViaCell, Inc.*, 491 F.3d 1342 (Fed. Cir. 2007).

Applicants respectfully submit that the pending rejection does not meet a *prima facie* case of obviousness, as described below.

#### *The Examiner’s Argument*

In the *Final Office Action* dated September 21, 2007, the Examiner, in rejecting claims 1, 13, 18, 21 and 26, cites to *Bays* as disclosing a computer-readable medium containing a program for managing annotations having multiple states which, when executed by a processor, performs operations comprising applying a set of state rules

to determine a first state of an annotation based on annotation data (citing to *Bays*, Col. 3, Lines 25-30). Further, the Examiner, in rejecting claims 1 and 21, also cites to *Bays* as disclosing applying the set of state rules to determine a second state of the annotation based on the annotation data in an updated annotation record (citing to *Bays*, Col. 9, lines 12-14, 59-65 and Col. 10, Lines 12-17).

*Applicants' Response to the Examiner's Argument*

Applicants submit that neither *Bays*, nor *Setya*, alone or in combination, teach or suggest the claim limitations of “applying a set of state rules to determine a first state of an annotation based on annotation data” and “applying the set of state rules to determine a second state of the annotation based on the annotation data in the updated annotation record.” Regarding the first claim limitation, as stated above, the Examiner argues that *Bays* discloses this limitation at Col. 3, Lines 25-30. The cited passage is provided below:

Filtering and transforming the entered annotation content based on the context of the read can be used to retrieve only relevant information, or to “hide” information to which this reader context is unauthorized, or to present the information in a form easily understood by the discipline or role of the reader. *Bays*, Col. 3, Lines 25-30.

Simply stated, the cited passage discloses a way to define what data to present to a user (*i.e.*, filtering), and how to present that data (*i.e.* transforming). Accordingly, Applicants submit that the cited passage has no relation to determining states of annotations.

Regarding the second claim limitation, as stated above, the Examiner argues that *Bays* discloses this limitation at Col. 9, lines 12-14, 59-65 and Col. 10, Lines 12-17. The cited passages are provided below:

In the example of FIG. 2, the administrator 27 defines the annotation structure by identifying the annotation structure by identifying the desired categories and the order in which the annotation content will be entered and/or displayed. *Bays*, Col. 9, 11-14.

The template transforming loop can be automatically initiated by the method 100 at the decision block 155, whereby, the method 100 inquires whether the administrator 27 (or application 22) wishes to specify a filter or modify a template to reflect the reader's context. The administrator 27 indicates which categories are to be retained, which attributes within these categories are to be retained, which attributes within these categories are to be retained, which attribute names are to be changed and how, and more generally transformations that should be applied to the annotation content. *Bays*, Col. 9, 56-65.

If the administrator 27 determines at block 170 that a template for an additional reader context is desired, the method 100 proceeds to block 160 and repeats the reader selection loop comprised of steps 160, 165, 170, until the administrator 27 determines at block 170 that no additional templates are desired. *Bays*, Col. 10, Lines 12-17.

Applicants submit that the cited passages simply provide greater detail to the transforming and filtering functionality disclosed in Col. 3, Lines 25-20 of *Bays*. Thus, the cited passages do not disclose anything with regard to determining a second state of annotation, let alone determining a first state of an annotation.

Therefore, *Bays* does not teach or suggest, “applying a set of state rules to determine a first state of an annotation based on annotation data” and “applying the set of state rules to determine a second state of the annotation based on the annotation data in the updated annotation record,” as suggested by the Examiner.

Thus, Applicants submit that neither *Bays*, nor *Setya*, alone or in combination, teach or suggest the claim limitations, and respectfully request the rejection of independent claims 1, 13, 18, 21, 26, and the claims that depend therefrom, be reversed and the claims be allowed.

## CONCLUSION

The Examiner errs in finding that claims 1, 4-21 and 34-31 are unpatentable over *Bays* in view of *Setya* under 35 U.S.C. § 103(a).

Withdrawal of the rejection and allowance of all claims is respectfully requested.

Respectfully submitted, and  
**S-signed pursuant to 37 CFR 1.4,**

/Gero G. McClellan, Reg. No. 44,227/

---

Gero G. McClellan  
Registration No. 44,227  
Patterson & Sheridan, L.L.P.  
3040 Post Oak Blvd. Suite 1500  
Houston, TX 77056  
Telephone: (713) 623-4844  
Facsimile: (713) 623-4846  
Attorney for Appellant(s)

## CLAIMS APPENDIX

1. (Previously Presented) A method for tracking the status of an annotation, comprising:
  - creating an annotation record comprising one or more fields for storing annotation data comprising the annotation;
  - retrieving annotation data stored in the annotation record;
  - applying a set of state rules to determine a first state of the annotation based on the annotation data; and
  - receiving additional annotation data;
  - updating the annotation record with the additional annotation data; and
  - applying the set of state rules to determine a second state of the annotation based on the annotation data in the updated annotation record;
  - providing an indication that the state of the annotation has changed from the first state to the second state.
- 2-3. (Cancelled)
4. (Previously Presented) The method of claim 1, wherein providing the indication that the state of the annotation has changed comprises notifying an entity.
5. (Original) The method of claim 4, wherein notifying the entity comprises notifying an entity responsible for providing annotation data required to change the state of the annotation from the second state to a third state.
6. (Original) The method of claim 1, wherein:
  - the annotation record is associated with a particular annotation type; and
  - the method further comprises retrieving the state rules based on a type of the annotation.
7. (Original) The method of claim 6, wherein the method further comprises:



generating an interface screen based on an annotation structure associated with the annotation type, wherein the annotation structure identifies the one or more fields;  
and

receiving annotation data via the interface.

8. (Original) The method of claim 1, wherein the annotation record comprises a field indicating the current state of the annotation.

9. (Original) The method of claim 1, further comprising initiating a process in response to detecting a change in the annotation state.

10. (Original) The method of claim 9, wherein the process examines a number of annotations in the same annotation state.

11. (Original) The method of claim 9, wherein the process modifies the annotation data.

12. (Original) The method of claim 9, wherein the process initiates another process.

13. (Original) A method for managing annotations having multiple states, comprising:

defining a plurality of annotation types, each annotation type having one or more associated fields;

defining a set of state rules for each annotation type, wherein each state rule identifies an annotation state based on annotation data in the one or more fields associated with its corresponding annotation type; and

providing a state machine capable of retrieving annotation data for an annotation of one of the defined annotation types, applying the state rules for that type to the annotation data to determine the state of the annotation, and providing an indication of the annotation state.

14. (Original) The method of claim 13, wherein at least one of the state rules for at least one of the annotation types identifies a state based on the presence or absence of data in at least one of the fields associated with that annotation type.

15. (Original) The method of claim 14, wherein at least one of the state rules for at least one of the annotation types identifies a state based on the presence or absence of data in at least two of the fields associated with that type.

16. (Original) The method of claim 13, wherein at least one of the state rules for at least one of the annotation types identifies a state based on a specified string of text in one of the fields associated with that type.

17. (Original) The method of claim 13, further comprising:  
    providing annotation structures for each annotation type, wherein each annotation structure identifies the one or more fields for a corresponding annotation type; and  
    generating annotation forms, based on the annotation structures, for receiving annotation data for each annotation type.

18. (Original) A method for gathering information about a plurality of processes of a similar process type, comprising:  
    providing an annotation form for receiving annotation data in a plurality of fields related to the processes;  
    storing annotation data received via the annotation form in a plurality of annotation records, wherein each annotation record relates to one of the similar type processes;  
    providing a set of state rules defining a plurality of states for the annotation based on the annotation data in each record;  
    applying the state rules to the annotation data in each record to determine the state of each annotation; and  
    generating a report indicating the state of each annotation.

19. (Original) The method of claim 18, wherein each annotation record comprises a field for storing the current state of the corresponding annotation.

20. (Original) The method of claim 18, wherein at least one of the state rules defines an annotation state based on presence or absence of data in one of the fields.

21. (Previously Presented) A computer-readable medium containing a program for managing annotations having multiple states which, when executed by a processor, performs operations comprising:

- creating an annotation record comprising one or more fields for storing annotation data;

- retrieving annotation data stored in the annotation record;

- applying a set of state rules to determine a first state of the annotation based on the annotation data;

- providing an indication of the first state of the annotation in an interface screen displaying the annotation data;

- receiving additional annotation data;

- updating the annotation record with the additional annotation data;

- applying the set of state rules to determine a second state of the annotation based on the annotation data in the updated annotation record; and

- providing an indication of the second state of the annotation in an interface screen displaying the annotation data.

22-23. (Cancelled)

24. (Original) The computer-readable medium of claim 21, further comprising notifying an entity of the state of the annotation.

25. (Original) The computer-readable medium of claim 24, wherein the entity is responsible for providing annotation data required to change the state of the annotation.

26. (Original) An annotation system, comprising:

- one or more annotation structures, each identifying one or more annotation fields associated with an annotation type;

an annotation store for storing annotation records, each having fields associated with one of the annotation types;

a set of state rules for each annotation type, wherein each set of state rules defines a plurality of states for each associated annotation type based on the annotation data in the one or more associated fields; and

a state machine configured to access an annotation record and apply the set of state rules for the corresponding annotation type to determine an annotation state based on the data stored therein.

27. (Original) The annotation system of claim 26, wherein the system further comprises an executable component for communicating a determined annotation state to an entity.

28. (Original) The annotation system of claim 27, wherein the executable component retrieves the annotation state as a field in an annotation record.

29. (Original) The annotation system of claim 27, wherein the executable component retrieves the annotation state from the state machine.

30. (Original) The annotation system of claim 26, wherein at least one of the state rules associated with one of the annotation types defines an annotation state based on the presence or absence of data in at least one of the associated fields.

31. (Original) The annotation system of claim 26, wherein at least one of the state rules associated with one of the annotation types defines an annotation state based on the presence or absence of a text string in at least one of the associated fields.

## EVIDENCE APPENDIX

None.

## RELATED PROCEEDINGS APPENDIX

None.